SERVLETS - SENDING EMAIL

http://www.tutorialspoint.com/servlets/servlets-sending-email.htm

To send an email using your a Servlet is simple enough but to start with you should have **JavaMail API** and **Java Activation Framework (JAF)** installed on your machine.

- You can download latest version of JavaMail (Version 1.2) from Java's standard website.
- You can download latest version of JAF (Version 1.1.1) from Java's standard website.

Download and unzip these files, in the newly created top level directories you will find a number of jar files for both the applications. You need to add **mail.jar** and **activation.jar** files in your CLASSPATH.

Send a Simple Email:

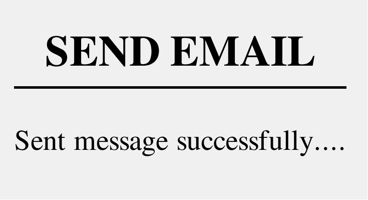
Here is an example to send a simple email from your machine. Here it is assumed that your **localhost** is connected to the internet and capable enough to send an email. Same time make sure all the jar files from Java Email API package and JAF package ara available in CLASSPATH.

```
// File Name SendEmail.java
import java.io.*;
import java.util.*;
import javax.servlet.*;
import javax.servlet.http.*;
import javax.mail.*;
import javax.mail.internet.*;
import javax.activation.*;

public class SendEmail extends HttpServlet{

  public void doGet(HttpServletRequest request,
                    HttpServletResponse response)
            throws ServletException, IOException
  {
      // Recipient's email ID needs to be mentioned.
      String to = "abcd@gmail.com";

      // Sender's email ID needs to be mentioned
      String from = "web@gmail.com";

      // Assuming you are sending email from localhost
      String host = "localhost";

      // Get system properties
      Properties properties = System.getProperties();

      // Setup mail server
      properties.setProperty("mail.smtp.host", host);

      // Get the default Session object.
      Session session = Session.getDefaultInstance(properties);

   // Set response content type
      response.setContentType("text/html");
      PrintWriter out = response.getWriter();

      try{
         // Create a default MimeMessage object.
         MimeMessage message = new MimeMessage(session);
         // Set From: header field of the header.
         message.setFrom(new InternetAddress(from));
         // Set To: header field of the header.
         message.addRecipient(Message.RecipientType.TO,
                                  new InternetAddress(to));
         // Set Subject: header field
         message.setSubject("This is the Subject Line!");
         // Now set the actual message
         message.setText("This is actual message");
         // Send message
         Transport.send(message);
         String title = "Send Email";
         String res = "Sent message successfully....";
         String docType =
         "<!doctype html public \"-//w3c//dtd html 4.0 " +
         "transitional//en\">\n";
         out.println(docType +
         "<html>\n" +
         "<head><title>" + title + "</title></head>\n" +
         "<body bgcolor=\"#f0f0f0\">\n" +
         "<h1 align=\"center\">" + title + "</h1>\n" +
         "<p align=\"center\">" + res + "</p>\n" +
         "</body></html>");
      }catch (MessagingException mex) {
         mex.printStackTrace();
      }
   }
}
```

Now let us compile above servlet and create following entries in web.xml

```
....
 <servlet>
     <servlet-name>SendEmail</servlet-name>
     <servlet-class>SendEmail</servlet-class>
 </servlet>

 <servlet-mapping>
     <servlet-name>SendEmail</servlet-name>
     <url-pattern>/SendEmail</url-pattern>
 </servlet-mapping>
....
```

Now call this servlet using URL http://localhost:8080/SendEmail which would send an email to given email ID *abcd@gmail.com* and would display following response:



SEND EMAIL

Sent message successfully....

If you want to send an email to multiple recipients then following methods would be used to specify multiple email IDs:

```
void addRecipients(Message.RecipientType type,
                   Address[] addresses)
throws MessagingException
```

Here is the description of the parameters:

- **type:** This would be set to TO, CC or BCC. Here CC represents Carbon Copy and BCC represents Black Carbon Copy. Example *Message.RecipientType.TO*

- **addresses:** This is the array of email ID. You would need to use InternetAddress() method while specifying email IDs

Send an HTML Email:

Here is an example to send an HTML email from your machine. Here it is assumed that your **localhost** is connected to the internet and capable enough to send an email. Same time make sure all the jar files from Java Email API package and JAF package ara available in CLASSPATH.

This example is very similar to previous one, except here we are using setContent() method to set content whose second argument is "text/html" to specify that the HTML content is included in the message.

Using this example, you can send as big as HTML content you like.

```
// File Name SendEmail.java
import java.io.*;
import java.util.*;
import javax.servlet.*;
import javax.servlet.http.*;
import javax.mail.*;
import javax.mail.internet.*;
import javax.activation.*;

public class SendEmail extends HttpServlet{

  public void doGet(HttpServletRequest request,
                    HttpServletResponse response)
            throws ServletException, IOException
  {
      // Recipient's email ID needs to be mentioned.
      String to = "abcd@gmail.com";

      // Sender's email ID needs to be mentioned
      String from = "web@gmail.com";

      // Assuming you are sending email from localhost
      String host = "localhost";

      // Get system properties
      Properties properties = System.getProperties();

      // Setup mail server
      properties.setProperty("mail.smtp.host", host);

      // Get the default Session object.
      Session session = Session.getDefaultInstance(properties);

   // Set response content type
      response.setContentType("text/html");
      PrintWriter out = response.getWriter();

      try{
         // Create a default MimeMessage object.
         MimeMessage message = new MimeMessage(session);
         // Set From: header field of the header.
         message.setFrom(new InternetAddress(from));
         // Set To: header field of the header.
         message.addRecipient(Message.RecipientType.TO,
                                  new InternetAddress(to));
         // Set Subject: header field
         message.setSubject("This is the Subject Line!");

         // Send the actual HTML message, as big as you like
         message.setContent("<h1>This is actual message</h1>",
                            "text/html" );
         // Send message
         Transport.send(message);
         String title = "Send Email";
         String res = "Sent message successfully....";
         String docType =
         "<!doctype html public \"-//w3c//dtd html 4.0 " +
         "transitional//en\">\n";
         out.println(docType +
         "<html>\n" +
         "<head><title>" + title + "</title></head>\n" +
         "<body bgcolor=\"#f0f0f0\">\n" +
         "<h1 align=\"center\">" + title + "</h1>\n" +
         "<p align=\"center\">" + res + "</p>\n" +
         "</body></html>");
      }catch (MessagingException mex) {
         mex.printStackTrace();
      }
   }
}
```

Compile and run above servlet to send HTML message on a given email ID.

Send Attachment in Email:

Here is an example to send an email with attachment from your machine. Here it is assumed that your **localhost** is connected to the internet and capable enough to send an email.

```
// File Name SendEmail.java
import java.io.*;
import java.util.*;
import javax.servlet.*;
import javax.servlet.http.*;
import javax.mail.*;
import javax.mail.internet.*;
import javax.activation.*;

public class SendEmail extends HttpServlet{

  public void doGet(HttpServletRequest request,
                    HttpServletResponse response)
            throws ServletException, IOException
  {
      // Recipient's email ID needs to be mentioned.
      String to = "abcd@gmail.com";

      // Sender's email ID needs to be mentioned
      String from = "web@gmail.com";

      // Assuming you are sending email from localhost
      String host = "localhost";

      // Get system properties
      Properties properties = System.getProperties();

      // Setup mail server
      properties.setProperty("mail.smtp.host", host);

      // Get the default Session object.
      Session session = Session.getDefaultInstance(properties);

   // Set response content type
      response.setContentType("text/html");
      PrintWriter out = response.getWriter();

       try{
         // Create a default MimeMessage object.
         MimeMessage message = new MimeMessage(session);

         // Set From: header field of the header.
         message.setFrom(new InternetAddress(from));

         // Set To: header field of the header.
         message.addRecipient(Message.RecipientType.TO,
                                  new InternetAddress(to));

         // Set Subject: header field
         message.setSubject("This is the Subject Line!");

         // Create the message part
         BodyPart messageBodyPart = new MimeBodyPart();

         // Fill the message
         messageBodyPart.setText("This is message body");

         // Create a multipar message
         Multipart multipart = new MimeMultipart();

         // Set text message part
         multipart.addBodyPart(messageBodyPart);

         // Part two is attachment
         messageBodyPart = new MimeBodyPart();
         String filename = "file.txt";
         DataSource source = new FileDataSource(filename);
         messageBodyPart.setDataHandler(new DataHandler(source));
         messageBodyPart.setFileName(filename);
         multipart.addBodyPart(messageBodyPart);

         // Send the complete message parts
         message.setContent(multipart );

         // Send message
         Transport.send(message);
         String title = "Send Email";
         String res = "Sent message successfully....";
         String docType =
         "<!doctype html public \"-//w3c//dtd html 4.0 " +
         "transitional//en\">\n";
         out.println(docType +
         "<html>\n" +
         "<head><title>" + title + "</title></head>\n" +
         "<body bgcolor=\"#f0f0f0\">\n" +
         "<h1 align=\"center\">" + title + "</h1>\n" +
         "<p align=\"center\">" + res + "</p>\n" +
         "</body></html>");
      }catch (MessagingException mex) {
         mex.printStackTrace();
      }
   }
}
```

Compile and run above servlet to send a file as an attachement along with a message on a given email ID.

User Authentication Part:

If it is required to provide user ID and Password to the email server for authentication purpose then you can set these properties as follows:

```
props.setProperty("mail.user", "myuser");
props.setProperty("mail.password", "mypwd");
```

Rest of the email sending mechanism would remain as explained above.